Deep Learning for Cyber Threat Detection: Challenges and Opportunities

Muntasir Maruf[1], Jannat Ara Tasnim[2], Hasin Aabrar Khan[3], Fatema Akther[4], Md Tanvir Bin Zoha[5], Rajarshi Roy Chowdhury[6]

Department of Computer Science
American International University-Bangladesh
Dhaka, Bangladesh

Abstract: The advancement of the Internet has made global communication more convenient. With the increasing demand for interconnected networks, the threats related to data privacy are also greater than ever before. Detecting cyber threats is becoming increasingly challenging as attackers continue to develop sophisticated techniques, including viruses, worms, ransomware, Denial of Service (DoS) attacks, and Structured Query Language (SQL) injection. The introduction of Deep Learning (DL) in cyber threat detection has opened a new opportunity to identify these threats before they exploit user security. This study analyzes the performance and limitations of several DL models including Deep Belief Network (DBN), Convolutional Neural Network Long Short-Term Memory Networks (CNN-LSTM) and hybrid models along with some IDSs (Intrusion Detection Systems) such as ASCHIDS (Adaptively Supervised and Clustered Hybrid Intrusion Detection System IDS) and RBC-IDS (Restricted Boltzmann Machine-based Clustered IDS. They showed promising results detecting these types of cyber threats. The primary contribution of this study is to provide some suggestions regarding the existing challenges of DL implementations, for example, dataset limitations, interpretability, optimization, scalability, and adaptability issues. This will enhance user safety by enabling more secure and resilient internet experiences.

Keywords: Cybersecurity, Cyber Threat, Threat Detection, Network Intrusion, Denial of Service, Deep Learning (DL), Intrusion Detection Systems (IDSs).

1. Introduction

Cyber threats can be regarded as hostile actions directed to compromise the integrity of a computing system, network or electronic devices connected to the network. For example unauthorized access, data breaches and suspicious activities along with various aggressive attacks such as phishing attacks that trick users into giving away their sensitive details [1], malware such as viruses that attach to a program or file and spread upon execution, stealing data and causing damage to files and others [2], worms that also damages or steals data but can replicate and spread automatically [2], ransomware whereas hackers hacks into the system encrypting the data and demands ransom for the encryption key [3], and trojans which comes with a genuine software to violates data integrity [2], as well as Denial of Service (DoS) or Distributed Denial of Service (DDoS) attacks that cripple systems by flooding the systems to great extents [4]. In addition, cyber attackers can use zero-day exploits in which attackers take advantage of unsolved flaws of a program since its release [5]. Structured Query Language (SQL) injection to access databases and exploit software weaknesses through injecting malicious queries through the input requests [2]. Reconnaissance is another type of attack where the systems and networks are monitored or surveyed to gather vulnerability details for exploitation [2]. Attacks, such as reconnaissance attempts to find system vulnerabilities. Whereas Remote-to-Local (R2L) attacks enable the remote user to get local access, User-to-Root (U2R) attacks take advantage of privilege escalation. Backdoors are hidden entrances that circumvent authentication. The Fuzzer tests the software by supplying malformed inputs that could lead to a crash or expose a defect. The path manipulation opens the way for directory traversal to gain access to protected files, Remote File Inclusion (RFI) forces the server to execute malicious files, buffer overflows overwrite memory to execute arbitrary code, and Cross-Site Scripting (XSS) inserts malicious code into web pages [2]. There could be many motives behind these attacks such as espionage, activism, profit or the mere desire of the attackers to create trouble [6]. Firewalls and antivirus software can be embraced to ensure safety from these types of threats. As well as strong password adoption, patching systems and

alertness on suspicious online activity for individual and corporate users also provides guard against cyber assault [7].

This study highlights the growing adoption of DL in cybersecurity, where it is being utilized to effectively analyze and mitigate sophisticated threats in real time. DL models comprehend data exponentially better at resolving associated complexities including malware, phishing and intrusions, when given massive amounts of data. To develop more precise detection of threats, response, DL models have the potential. The DL market has generated amazing revenues and profits over the years as a nascent technology in various industries. The market was put at USD 96.8 billion in 2024 and was estimated to increase at a compound annual growth rate or Compounded Annual Growth Rate (CAGR) of 31.8% to reach approximately USD 526.7 billion in 2030 [8]. The growth is prompted by increased computing power, the big data boom and the deployment of DL solutions in diverse industries such as health, finance, and manufacturing.

Although DL has made great strides, it still faces many problems. The most important of these is the "vanishing gradient problem": the gradients in training neural networks become so small that learning is impeded in deep architectures. Most of them require massive volumes of labeled data which are usually very expensive to acquire [9]. Another important concern is that DL models are black boxes. This means that it is very challenging to know how they make decisions [10].

The Internet has indeed revolutionized global communication, yet that comes with threats to intrusion upon the data privacy. Detecting cyber threats poses tougher challenges against threats like viruses, worms, ransomware, and DoS attacks. The introduction of DL might be proven helpful to solve this emerging issue. This study analyzes DL models and DL based solutions in order to compare performance and highlight existing gaps and possible solutions into limitations posed by datasets and adaptability. The contribution of the study as follows-

- Gather insights on the DL approaches for network/cyber intrusion detection. Several research papers have been studied to know about the current state of cyber threat detection using DL.
- Analyse the performance of several DL model implementations for detecting common cyber threats and attacks.
- Understand the outcomes and challenges faced by the researchers during implementation and find the limitations of their research.

The remainder of this study is organized as follows. Section 2 outlines the journal/article selection and review methodology employed in this study. This section reveals also the DL models used for threat detection and the datasets applied in this domain. In Section 3 key findings and limitations extracted from the selected paper are discussed. Section 4 identifies the major challenges faced by researchers during the implementation of DL models and highlights potential research opportunities. Section 5 represents several recommendations aimed at improving detection performance. Finally, Section 6 draws the conclusion and provides future direction.

2. Review Process

In this study Systematic Literature Review (SLR) is used to gather information regarding cyber threat detection and the usage of DL in this field. Research databases such as IEEE Explore, SpringerLink, Science Direct, MDPI, ACM, and Wiley were used as a source for searching related research papers. Several key phrases, such as "Cyber threat detection using DL", were used to narrow down the search field. Altogether, a list of 25 initially selected papers was prepared. Figure 1 illustrates the entire paper selection process from initial selection to final selection.

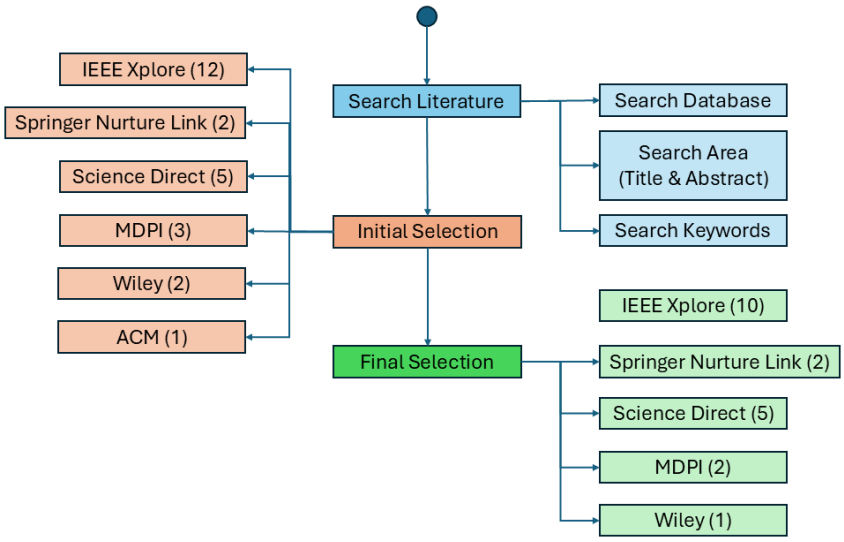

Figure 1: Paper selection process

Models/Approaches

Among all the DL models and techniques, some have exhibited promising results in the detection of cyber threats. The fundamental concepts underlying the models are as follows-

- Convolutional Neural Networks (CNNs): CNN is a type of Neural Network (NN) learn spatial patterns by convolutional layers that are specially designed for image-processing tasks. This model is mostly used for extracting spatial patterns from the data and detects classes based on structured input features and widely used for image and video recognition [11].
- Recurrent Neural Networks (RNNs): These are well-fitted for sequential data tasks as language modeling, speech recognition, and time-series forecasting [12].
- Deep Neural Networks (DNNs): DNNs are multilayer networks that are used for the recognition of very complex patterns in information. These models can be used for classification, regression, and pattern recognition across various domains [13].
- Transformer Models: These types of models parallelize the processing of their input sequences while using attention mechanisms in their natural language processing and other environments like cyber anomaly detection [14].

Figure 2 represents the basic structure of a neural network. Consisting Sigmoid activation in each node and SoftMax at the output nodes.

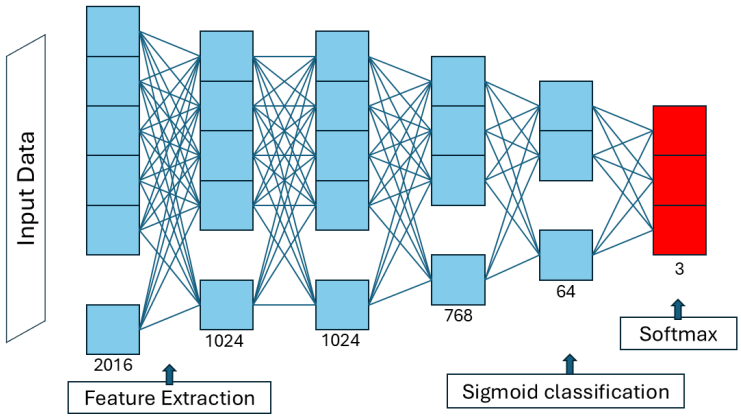

Figure 2: An abstract design of the basic NN architecture

- Long Short-Term Memory Networks (LSTM): An improved RNN that solve vanishing gradient problems and can learn long-term dependencies more readily. This feature is useful for language modeling, machine translation, and time-series prediction [15]. Figure 3 illustrates LSTM featuring short-term and long-term memory and a forget irrelevant information feature to improve model performance.

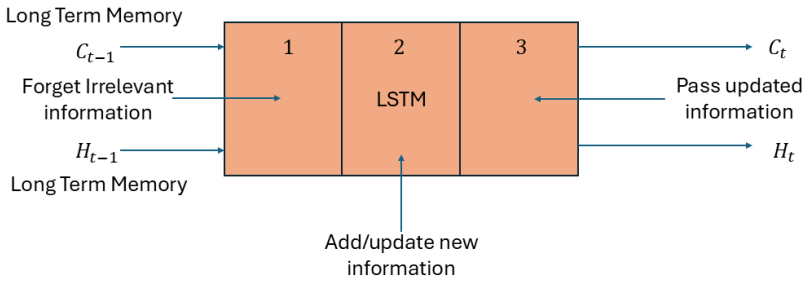

Figure 3: An abstract design of the LSTM architecture

- Deep Belief Networks (DBN): Unlike the other models mentioned in this section, the generative model for unsupervised learning consists of many layers of Restricted Boltzmann Machines [16]. Figure 4 represents the basic structure of DBN where multiple RBMs are stacked together to create a neural network.

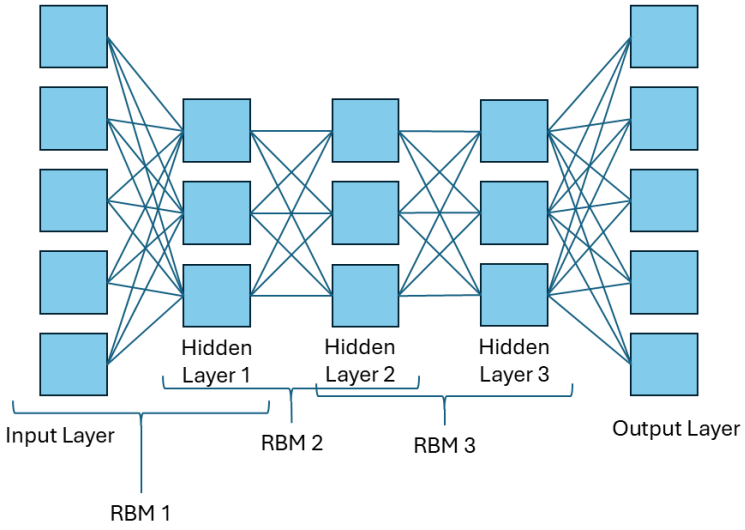

Figure 4: An abstract design of the DBN architecture

- Graph Neural Networks (GNN): Designed for processing graph data, these networks can be utilized in social network analysis [17].
- Generative Adversarial Network (GAN): It is a DL architecture that trains two neural networks to compete against each other to generate more authentic new data from a given training dataset. Particularly useful for generating synthetic data with high level of accuracy and precision [18]. Figure 5 demonstrates the general structure of a GAN. Where the generator model and the discriminator model work sequentially.

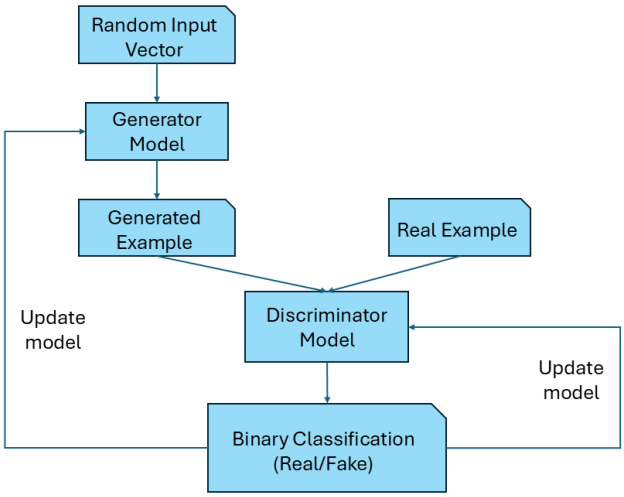

Figure 5: An abstract design of the GAN architecture

- Gradient Boosting Trees (GBT): This is a powerful method mainly used for structured data classification and regression. The concept is to build models one by one, with each new model correcting the errors of the previous ones [19].
- Application Specific Integrated Circuits (ASIC) is a specialized functional processor designed to target a specific task. This usually offers better performance than general-purpose processors [20].
- Field Programmable Gate Arrays (FPGA) are programmable processors speeding up operations of the DL [21].
- Network Intrusion Detection Systems (NIDS) offers real-time network intrusion monitoring [22].
- Host Intrusion Detection System (HIDS) monitors and analyses system activities to detect malicious behaviour [23].
- ASCH-IDS (Adaptively Supervised and Clustered Hybrid Intrusion Detection System) is designed for wireless networks where detection is done via feature distribution across layers of clustered nodes for improved accuracy [24].
- RBC-IDS (Restricted Boltzmann-based Clustered Intrusion Detection System) is also designed for wireless but uses predefined rules and performs better than ASCH- IDS [25].
- Explainable Artificial Intelligence (XAI) focuses on the interpretability and transparency of AI models, removing the black box nature to ensure fairness, trust, and further improvements. SHAP (Shapley Additive Explanations) and LIME (Local Interpretable Model-agnostic Explanations) are the most popular methods [26].
- Ensemble Models are combination of multiple models and techniques.

Datasets

To properly train, test, and evaluate the models and mechanisms, a suitable dataset is a strong requirement. Use of balanced, adequate, and quality datasets with proper model selection can drastically improve the detection performance [27]. Here are some important datasets used by researchers in their studies.

- Knowledge Discovery in Databases (KDD'99): Derived from the 1998 DARPA (Defence Advanced Research Projects Agency) intrusion detection evaluation program, it was one of the first publicly available datasets for IDS research. Contains network connection labelled as

normal or several attack types, including DoS, Probe, Remote to Local (R2L), and User to Root (U2R) [28].

- Network Security Laboratory - Knowledge Discovery in Database (NSL-KDD): An improved version of the KDD'99 dataset. NSL-KDD removes redundant records to address the bias and imbalance problems in the original KDD'99 [29].
- University of New South Wales Network Behaviour 2015 (UNSW-NB15): Developed by the Australian Centre for Cyber Security. It includes modern attack types such as Fuzzers, Backdoors, DoS, Exploits, Reconnaissance, Worms, etc. It contains raw network traffic and label, indicating normal and malicious behaviour [30]
- Canadian Institute for Cybersecurity – Intrusion Detection System 2017 (CICIDS2017): Captures realistic network traffic based on user profiles, including benign and various attack scenarios over a five-day period. Attacks include DDoS, Brute force, Botnet, Port scan, Web attacks, Infiltration, etc. [31].
- Aegean Wireless Intrusion Detection (AWID): A dataset developed by the University of Aegean. It is designed using wireless network traffic in real-world scenarios to detect attacks such as authentication, flooding, injection etc. via wireless methods [32].
- Centro Superior de Investigaciones Científicas 2010 HTTP (CSIC-2010): The Spanish National Research Council created the labelled CSIC-2010 HTTP dataset to assess intrusion detection systems that target web application attacks. Thousands of manually created malicious and legitimate HTTP requests are included [33].
- Botnet – Internet of Things (Bot-IoT): A dataset designed for DDoS, DoS, data theft, and reconnaissance attacks on IoT environments, covering botnet activities. Generated using real-world tools on a testbed [34].

Table 1 provides a breakdown analysis of the datasets used in this research, referencing their name, contents, and which types of cyber threats they can represent, and Table 2 provides a comparative review of various deep learning methods applied to network intrusion detection, showing their performance on a variety of benchmark datasets.

Table 1: Overview of the datasets

Name	Overview	Detectable Threats	Source
KDD'99	Labeled records of normal traffic and various cyberattacks.	DoS, Probing, R2L, U2R attacks.	[28]
NSL-KDD	Refined version of KDD'99 with reduced redundancy and improved class balance.	DoS, Probing, R2L, U2R attacks.	[29]
UNSW-NB15	Modern network traffic behaviors. Combination of normal and diverse attack types in a realistic environment.	Fuzzing, Backdoors, DoS, Exploits, Reconnaissance, Worms,Shellcode, Generic attacks	[30]
CICIDS-2017	Simulated real-world network traffic generated over several days.	DDoS, Brute Force, Botnet, Port Scanning, Web Attacks, Infiltration	[31]
AWID	Wireless network traffic dataset collected in real-world environments.	Authentication Bypass, Flooding, Injection attacks (via wireless networks)	[32]

CSIC-2010	Contains manually generated HTTP request logs simulating both legitimate (normal) and malicious (attack) requests.	SQL Injection, XSS, buffer overflows, directory traversal, remote file inclusion, SQL injection, and DoS.	[33]
Bot-IoT	Labeled data reflecting both normal behavior and IoT-specific attack scenarios.	DDoS, DoS, Data Theft, Reconnaissance (Mainly IoT-related cyber threats)	[34]

Table 2: Performance overview of the models/approaches

Source	DL Models/Approaches	Dataset	Performance
[35]	CNN, RNN, Autoencoders	NSL-KDD	CNN 99.2% ^{A}, RNN 98.7% ^{A}, Autoencoder 96.3% ^{A}
[36]	Stacked Autoencoders + Big Data Tools	NSL-KDD	98.4% ^{A}, 97.8% ^{P}, 98.1% ^{R}
[37]	CNN + RNN	NSL-KDD UNSW-NB15	99.2% ^{A}, 98.9% P, 99.1% R; On 98.7% ^{A}
[38]	LSTM, GRU, DNN	CICIDS	LSTM 98.5% ^{A}, GRU 98.3% ^{A}, DNN 97.8% ^{A}
[39]	DNN with wrapper-based feature selection	AWID	98.7% ^{A}
[40]	CNN-LSTM	CISC-2010	97.9% ^{A}, 96.8% ^{F1}
[41]	DDoSnet (custom CNN)	CICIDS-2017	99.6% ^{A}
[42]	GANs + CNN	NSL-KDD	92.3% ^{F1}
[43]	CNN-RNN	KDD Cup 99	98.5% ^{A}
[44]	CNN, RNN	Bot-IoT	97.2% ^{A}

Note: Accuracy – A, Precision – P, F1-Score – F, Recall – R

From the performance analysis represented in Table 2, it can be observed that methods such as CNN, RNN, LSTM, Autoencoders, and hybrids have all been tested across KDD, NSL-KDD, CISC, CICIDS, UNSW-NB15, AWID, and Bot-IoT datasets. Most of the models show a very high correctness percentage with numbers going as high as 97 in most cases. Other quality measures like precision, recall, and F1-score have also been provided for some of the methods. Interestingly, a custom CNN model named DDoSnet gave the highest accuracy of 99.6%, on the CICIDS-2017 dataset. This table further justifies that deep learning works best in the intrusion detection domain, with special emphasis on CNN and RNN based models.

3. Findings & Limitations

Table 3 highlights the findings observed while reviewing articles on the use of DL in the detection of cyber threats, along with the limitations faced by the authors.

Table 3: Findings and Limitations

Source	Findings	Limitations
[35]	Using 40% of the training data, DBN achieved 97.5% accuracy.	Limited to only **NSL-KDD dataset and** does not fully represent the complexity of modern real-world

	ASIC and FPGA techniques are expensive, inflexible, and not very scalable. Usage of Autoencoder and DBN, as a Hybrid Model, was better performing in terms of detection accuracy compared to a single DBN.	traffic. The results are limited to only accuracy and f1 score. Metrics such as precision, recall, confusion matrices, and Area Under the Curve (AUC)-Receiver Operating Characteristic (ROC) curves could better rep- resent the overall performance.
[25]	The DL model RBC-IDS and ML model ASCH-IDS both managed to achieve accuracy rates of 99% for intrusion detection. The training and testing time of the ASCH-IDS was almost half that of RBC-IDS.	The Dataset (KDD'99) may not fully represent the complexity of modern real-world traffic. Lacks testing under adverse network conditions or with real-time adaptive attacks.
[36]	For binary classification DNN has the highest accuracy (99.19%) on the UNSW-NB15 dataset. But ML GBT has the best accuracy (99.97%) on the CICIDS2017 dataset, along with the lowest prediction time. For multiclass classification DNN has the better accuracy (97.04%) for UNSW-NB15 dataset, and 99.57% for CICIDS2017 dataset.	Limited experimentation with feature selection for handling heterogeneous data. As multiclass classification is not supported by GBT, only DNN and RF are used for that. Unevaluated performance impact because of changing the number of computing nodes in the Spark cluster.
[45]	SHAP and LIME helped to identify the critical features like destination port number and source IP address, and to interpret decisions for specific instances. Usage of Autoencoder and DBN, as a Hybrid Model, provided a higher detection accuracy compared to a single DBN. The Fully Connected Network (FCN) model showed promising results over traditional ML models such as SVM (Support Vector Machine).	Limited exploration of alternative ensemble strategies (e.g., stacking or soft voting). More validation in real-world IoT environments is required under variable conditions. Scalability and computational efficiency in larger IoT systems were not fully addressed.
[46]	RNNs, CNNs and GNNs provide improved detection performance compared to traditional methods. GNNs help comprehend the structural relationships within an organization, whereas RNNs are especially good at modelling user activity sequences.	Over-reliance on synthetic datasets such as those of CERT (Computer Emergency Response Team), which do not account well for the complexity of actual insider behavior. Models currently available also lack interpretability, rendering it difficult for analysts to trust or understand what they produce.
[37]	DNN trained predictions can outperform classical machine learning predictions in detection accuracy and generalization. Strong results are reported from KDDCup99, NSL-KDD, UNSW-NB15, and CICIDS 2017. Presents the SHIA: Scale-Hybrid-IDS-AlertNet, a hybrid framework comprising	The system achieves a high-performance level but is not interpretable, making it difficult for any security analyst to understand the rationale for the alerts generated. The model also has very high computational demands and has not been rigorously tested for effectiveness in the real world.

	both NIDS and HIDS	
[47]	CNNs and RNNs had better performance than previous ML algorithms in malware classification and detection. Proposed ScaleMalNet is a highly scalable and hybrid architecture that brings together many detection approaches, incorporating self-learning techniques for detection, classification and categorization of threats and malware.	Use of private datasets that impinge on reproducibility. The model also has very high computational demands and has not been rigorously tested for effectiveness in the real world. The study is more concerned with classification accuracy, ignoring model interpretability and adversarial robustness.
[38]	LSTM, CNN, and Autoencoders have shown promise in enhancing intrusion detection rates and minimizing false alarms. DL algorithms can automatically extract complex features from raw data for improved detection in contrast to traditional ML approaches. Hybrid approaches mixing ML and DL, usually been better than their counterpart single model methods.	Minimal use of real-world datasets, almost no attention to interpretability of the model, and inadequate evaluation metrics aside from accuracy (e.g., precision, recall, AUC). Outdated or limited-scope datasets such as NSL-KDD. Limited research on making AI-based NIDS resilient to adversarial attacks.
[39]	Proposed a feature selection mechanism based on Extra Tree and fed into an FFDNN for classification purposes. The system performance at AWID was exceptional, yielding 99.66% for binary classification and 99.77% for multiclass performance, outperforming traditional ML models.	The method is computationally expensive and may not be optimal for lightweight or mobile devices. Generalization to other real-world cases may be affected due to the dependence on certain datasets. Lacks interpretability, adversarial robustness testing to ensure that it is not easy to manipulate by sophisticated attacks.
[40]	A CNN-LSTM architecture trained in a distributed manner across edge devices, achieved 94.2% detection accuracy on the CSIC 2010 dataset. Experiments show a high detection level and strong detection performance for web-based attack types, including SQL injection and cross-site scripting (XSS).	Experiments were conducted on a single dataset, CSIC 2010, which may not sufficiently represent modern attack patterns or the variations of real traffic. Lacks adaptability of the model to more dynamic settings, or with changing attack signatures. Unexplored effects of different hardware platforms, or network conditions, on detection performance, and exploits robustness in heterogeneous deployments.
[41]	DDoSNet is evaluated on the CICIDS2017 dataset, displaying impressive metrics such as accuracy, precision, recall, and F1-scores. The model employs both an autoencoder for feature extraction and LSTM for sequence modeling. Reported detection accuracy above 99% and a low false positive rate.	Reliance on labeled datasets, such as CICIDS2017, that may not reflect current and emerging modes of DDoS attacks in the wild. Does not discuss how the model may scale in very large or geographically distributed networks, where deployment constraints and latency could present operational challenges.
[42]	ML models such as Random Forest and XGBoost have a better performance than SVM and traditional DNN in classifying imbalanced network traffic. Ensemble models yield better precision	Used only static oversampling methods. Only uses one dataset (UNSW-NB15), which may limit the applicability of their finding. Does not consider other more sophisticated models,

	and recall for minority attack classes, such as Shellcode and Worms. Supports oversampling using SMOTE (Synthetic Minority Oversampling Technique) plus Tomek Links, improving classifier performance.	such as CNN-LSTM. Which might perform better. Unconsidered implications of real-time use and model drift.
[43]	CNN-LSTM model has better detection accuracy and efficiency than traditional ML models and standalone DL architecture. With the NSL-KDD dataset, the proposed system obtains higher performance metrics with respect to the detection rate and the false alarm rate. The hybrid model is especially good at recognizing the U2R (User to Root) and R2L (Remote to Local) attack classes.	NSL-KDD dataset is an old benchmark dataset, and unlikely to reflect more recent or advanced attacks. Lacks discussion of computational costs for model evaluation, training time resource utilization for either model learning or inference and related metrics.
[48]	Ensemble model, Combining CNN and LSTM, significantly outperforms individual DL models like CNN or LSTM alone. Tested on the Secure Water Treatment (SWaT) dataset, a well-known real-world ICS dataset, achieving an accuracy of 99.98%, along with high precision, recall, and F1-score.	Use of only the SWaT dataset limits its generalizability to other ICS environments. Its performance on different systems or datasets is unknown. Leading to adaptability concerns. Lacks explainability. DL ensembles act as black boxes, offering little insight into how decisions are made.
[49]	Highlights the superiority of DL models, particularly CNNs and deep autoencoders (DAs), over traditional machine learning methods for intrusion detection. Using real traffic datasets (CSE-CIC- IDS2018 and Bot-IoT). CNNs achieved the highest detection rate on CSECIC IDS2018, while DAs excelled on Bot-IoT.	The evaluation metrics used, like accuracy and detection rate, do not fully capture other important factors like computational complexity, scalability, and adaptability to evolving threats. High computational cost of DL models. Lack of real-time detection and response solutions and detailed comparison of models in various contexts.
[44]	CNN+LSTM hybrid model reached a top accuracy of 97.16% and a recall of 99.1%, significantly surpassing all other models. DL models, especially LSTM and CNN+LSTM works exceptionally well for detecting DDoS attacks in IoT settings.	Lengthy training time, particularly with extensive datasets, which hinders model development. Balancing the original unbalanced dataset required manual balancing by duplication, which might have led to redundancy or bias. Although DL models eliminate the need for manual feature selection, having too many attributes can impede efficiency, indicating that feature selection may still be beneficial.
[50]	DL models like DNN and RNN often performed better in detecting anomalies compared to traditional ML methods. The paper reviews important datasets like KDD Cup 1999, ISOT, HTTP CSIC 2010, and UNSW-NB15, outlining their usefulness and limitations in training and evaluating ML/DL models.	DL models demand high computational resources, making them difficult to implement for organizations with limited infrastructure. The black-box nature of ML/DL models further complicates their use in cybersecurity, where decision transparency is critical.
[51]	DL methods have outperformed traditional	Absence of large, openly available datasets that

	signature-based and rule-based systems in classifying and detecting cyberattacks. DL models acquire high true positive rates (TPR) in classifying malicious domain names (96.01–99.86%) and network intrusions (92.33–100%).	would make model development and comparison troublesome using DL. Black-box nature of DL models also raises issues about interpretability and trust, particularly in security applications where stakes are high.
[52]	Integrates an IoT device-embedded DCNN for application layer DDoS and phishing detection, and a cloud-based LSTM model for botnet attack detection. Superior performance with 94.3% accuracy and 93.58% F1-score for DCNN-based phishing detection, and 94.80% accuracy for LSTM-based botnet detection.	While the phishing, DDoS, and botnet attacks are very nicely addressed by this research work, other novel attacks on IoT remain unchecked. The effectiveness of the suggested framework in handling the encrypted traffic scenarios has never been tested in a real-world scalable environment.
[53]	Proposed DL-based IDS achieves good results for the detection of multiple attacks (e.g., blackhole, DDoS, sinkhole, and wormhole) in the IoT context. DNN model reconfiguring DBN achieved 95% in average precision and 97% in recall. IDS outperforms conventional methods in multiple attack instances, specifically in terms of precision and F1-score.	The practice of deploying this system, with respect to deploying the system in WLANs, will make the nature and importance of performance evaluation very different than that observed in a simulation. The current state of success against zero-day attacks being unknown and not tested, IES cannot be trusted.

From the findings and limitations listed in Table-3 it can be discussed that DL algorithms such as CNN, RNN, DNN, LSTM, and Autoencoder outclass usual ML techniques in attack detection. Hybrid and ensemble architectures such as CNN-LSTM and Autoencoder-DNN outperform simple forms of attack detection into DDoS, phishing, malware, U2R, and R2L detection. DL detection approaches work well at the real-time level and in IoT scenarios if implemented at an edge. Interpretability algorithms such as SHAP and LIME provide a backbone for increased understanding of model decisions, thus improving transparency and trustworthiness. Nonetheless, there are several difficulties with DL-based intrusion detection systems. Such as, dependency on artificially generated or outdated datasets, lack of extensive testing in real-world environments, and increased computational power. Moreover, most DL models act as a black box with respect to interpretability, thus becoming a problem in security-critical applications. Also, in some cases without including AUC, precision, recall, and robustness as standard evaluation metrics, only accuracy is heavily favored. Another problem is dataset imbalance, which leads to biased or meaningless results or subpar performance. These restrictions present several significant problems that require across-the-board solutions in order to guarantee scalability, stability, and usefulness.

4. Challenges & Opportunities

The research reveals that several challenges remain to realize DL's full potential in cyber threat detection. This provides excellent opportunities for further improving cyber threat detection based on DL, addressing the crucial constraints that hinder real-world use cases.

- A prominent challenge is to use updated real-world datasets, or dynamic data that updates with network activities. Because DL models still depend on outdated or synthetic datasets (e.g., NSL-KDD, KDD'99), which do not adequately reflect the complexity of today's network traffic and new attack methods [25], [35], [38], [40], [41], [42], [43]. Another

prevalent issue regarding datasets is data imbalance, which tends to create skewed models with very high false positive rates [44]. This brings forth an excellent opportunity for creating an upto date network intrusion dataset with balanced distribution.

- Consequently, some DL models fail to generalize and scale in an updated manner to provide a large-scale implementation [45], [48], [49]. Hence, improvement in generalization and scalability is still a challenge to overcome.
- Furthermore, deep learning models are profoundly computationally expensive and require significant resources to train and hence are difficult to deploy on low power or resource-constrained systems [39], [47], [49]. Therefore, creation of robust and ligth weight model/implementaion can be a very good opportunity.
- Interpretability is still a significant concern, as most DL models function as black boxes, revealing very little about how their decisions are made. This is especially troubling in security-critical applications [37], [39], [46], [50], [51]. A solution to make the models interpretable is required, allowing us to better understand and incorporate improvements.
- Traditional DL implementations lack adaptability because of the dynamic nature of cyber threats and attack signatures, meanwhile, most of the datasets are static [40], [48]. Hence, proper incorporation of adaptibility on DL based solutions is necessary.

5. Recommendations

To overcome the limitations identified in this study, several methodological advancements can be proposed to improve the performance of cyber threat detection using deep learning techniques. Such as-

- Using more realistic datasets: Most existing studies rely on synthetic and outdated datasets such as KDD'99 and NSL-KDD. These datasets do not represent modern, real-world network traffic. The performance of any DL model is greatly affected by the quality of the data that used to train the model [54]. Therefore, greater emphasis should be placed on the use of diverse, up-to-date, and publicly available.
- Implementation of XAI: DL models are often difficult to interpret and lead to trust issues in practical deployment. Integrating explainability techniques such as SHAP or LIME can provide model transparency and enhance analyst confidence [26].
- Real-Time Detection and Scalability: Real-time performance and scalability remain major challenges for researchers. Hence, it is essential to design models lightweight and efficient, that can be deployed in edge as well as cloud environments [55].
- Adversarial Resilience: Many existing approaches fail to adequately address adversarial robustness. Future work should include adversarial training and robustness evaluation to enhance resistance against evasion and manipulation attacks [56].
- Exploration of Ensemble Methods: Detection accuracy and flexibility can be improved through ensemble techniques such as stacking and soft voting. Integrating base models through innovative ensemble strategies can also yield performance gains.
- Cross-Dataset Validation and Hyperparameter Tuning: Applying rigorous cross-dataset validation and optimizing hyperparameters are essential steps toward improving model generalizability and efficiency in diverse network environments [57].
- Development of Hybrid Models: As this study reveals that the combination of multiple DL models and the incorporation of DL models with traditional rule-based systems can produce more robust hybrid models. Consequently, the development of additional hybridization techniques is imperative to achieve more robust and reliable solutions.

6. Conclusion

The research reviewed here indicates that integrating distributed DL architectures into edge computing can significantly enhance the privacy and efficiency of web attack detection systems. By decentralizing threat detection using CNN-LSTM models combined with blockchain-based consensus mechanisms, the proposed approach achieved high detection accuracy while addressing critical challenges related to data privacy, latency, and computational constraints. This method- ology demonstrates strong potential for practical deployment in latency-sensitive and privacy-critical environments, such as smart cities and industrial IoT applications. However, certain limitations remain, including reliance on a single dataset, the need for improved model generalization, and the challenge of hardware heterogeneity.

Future research should focus more on creating self-adaptive models that can recognize and react to new and changing cyberthreats. Emphasis should be placed on addressing data imbalance through advanced resampling techniques, domain-specific loss functions, or synthetic data generation to improve model robustness and accuracy. Constructing diverse and representative datasets that reflect various network environments and attack behaviors remains essential. Additionally, integrating multi modal data sources such as sensor inputs, traffic patterns, and contextual metadata can enhance situational awareness and detection precision. Further exploration of decentralized DL approaches, particularly federated learning, is also crucial to ensuring data privacy while maintaining scalability and low latency performance. These research directions will be central to advancing next generation intrusion detection systems that can effectively secure dynamic and heterogeneous network infrastructures.

References

[1] IBM, "What is phishing?," 2024. [Online]. Available: https://www.ibm.com/think/topics/phishing

[2] Ö. Aslan, S. S. Aktuğ, M. Ozkan-Okay, A. A. Yilmaz, and E. Akin, "A Comprehensive Review of Cyber Security Vulnerabilities, Threats, Attacks, and Solutions," *Electronics (Basel)*, vol. 12, no. 6, p. 1333, Mar. 2023, doi: 10.3390/electronics12061333.

[3] A. Gazet, "Comparative analysis of various ransomware virii," *Journal in Computer Virology*, vol. 6, no. 1, pp. 77–90, Feb. 2010, doi: 10.1007/s11416-008-0092-2.

[4] M. A. Obaidat, S. Obeidat, J. Holst, A. Al Hayajneh, and J. Brown, "A Comprehensive and Systematic Survey on the Internet of Things: Security and Privacy Challenges, Security Frameworks, Enabling Technologies, Threats, Vulnerabilities and Countermeasures," *Computers*, vol. 9, no. 2, p. 44, May 2020, doi: 10.3390/computers9020044.

[5] IBM, "What is a zero-day exploit?," 2025. [Online]. Available: https://www.ibm.com/think/topics/zero-day

[6] T. J. Holt *et al.*, "An exploratory analysis of the characteristics of ideologically motivated cyberattacks," *Terrorism and Political Violence*, vol. 34, no. 7, pp. 1305–1320, 2022.

[7] J. M. Borky and T. H. Bradley, "Protecting Information with Cybersecurity," in *Effective Model-Based Systems Engineering*, Springer, 2018, pp. 345–404. doi: 10.1007/978-3-319-95669-5_10.

[8] "Deep Learning Market Size, Share & Trends Analysis Report by Hardware, by Software, by Application, by End Use, by Region, and Segment Forecasts, 2023 - 2030," 2024. [Online]. Available: https://www.grandviewresearch.com/industry-analysis/deep-learning-market

[9] "The Vanishing Gradient Problem in Deep Learning: Causes, Effects, and Solutions," 2025. [Online]. Available: https://www.lunartech.ai/blog/the-vanishing-gradient-problem-in-deep-learning-causes-effects-and-solutions

[10] "AI's Mysterious 'Black Box' Problem, Explained," 2023. [Online]. Available: https://umdearborn.edu/news/ais-mysterious-black-box-problem-explained

[11] Manav, "Convolutional Neural Networks (CNN) in Deep Learning," 2025. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/

[12] Manav, "A Brief Overview of Recurrent Neural Networks (RNN)," 2022. [Online]. Available: https://www.analyticsvidhya.com/blog/2022/03/a-brief-overview-of-recurrent-neural-networks-rnn/

[13] M. Mercier, "What is a Deep Neural Network?," 2025. [Online]. Available: https://botpress.com/blog/deep-neural-network

[14] C. Stryker and D. Bergmann, “What is a Transformer Model?,” 2025. [Online]. Available: https://www.ibm.com/think/topics/transformer-model
[15] S. Saxena, “What is LSTM? Introduction to Long Short-Term Memory,” 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/03/introduction-to-long-short-term-memory-lstm/
[16] D. Kalita, “An Overview of Deep Belief Network (DBN) in Deep Learning,” 2024. [Online]. Available: https://www.analyticsvidhya.com/blog/2022/03/an-overview-of-deep-belief-network-dbn-in-deep-learning/
[17] A. Pearce, A. Wiltschko, B. Sanchez-Lengeling, and E. Reif, “A Gentle Introduction to Graph Neural Networks,” 2021. [Online]. Available: https://distill.pub/2021/gnn-intro/
[18] S. Robinson, K. Yasar, and S. Lewis, “What is a Generative Adversarial Network (GAN)?,” 2024. [Online]. Available: https://www.techtarget.com/searchenterpriseai/definition/generative-adversarial-network-GAN
[19] Gaurav, “An Introduction to Gradient Boosting Decision Trees,” 2021. [Online]. Available: https://www.machinelearningplus.com/machine-learning/an-introduction-to-gradient-boosting-decision-trees/
[20] “Application-Specific Integrated Circuit,” 2025. [Online]. Available: https://www.sciencedirect.com/topics/computer-science/application-specific-integrated-circuit
[21] A. Boutros, A. Arora, and V. Betz, “Field-Programmable Gate Array Architecture for Deep Learning: Survey & Future Directions,” 2024. [Online]. Available: https://arxiv.org/abs/2404.10076
[22] Sapphire, “What is Network Intrusion Detection System (NIDS)?,” 2023. [Online]. Available: https://www.sapphire.net/blogs-press-releases/nids/
[23] Redscan, “Host-Based Intrusion Detection (HIDS),” 2025. [Online]. Available: https://www.redscan.com/services/hids/
[24] S. Otoum, B. Kantarci, and H. T. Mouftah, “Adaptively Supervised and Intrusion-Aware Data Aggregation for Wireless Sensor Clusters in Critical Infrastructures,” in *2018 IEEE International Conference on Communications (ICC)*, IEEE, 2018, pp. 1–6. doi: 10.1109/ICC.2018.8422528.
[25] S. Otoum, B. Kantarci, and H. T. Mouftah, “On the feasibility of deep learning in sensor network intrusion detection,” *IEEE Networking Letters*, vol. 1, no. 2, pp. 68–71, 2019.
[26] IBM, “Explainable AI,” 2025. [Online]. Available: https://www.ibm.com/think/topics/explainable-ai
[27] A. Bailly *et al.*, “Effects of dataset size and interactions on the prediction performance of logistic regression and deep learning models,” *Comput Methods Programs Biomed*, vol. 213, p. 106504, 2022.
[28] U. C. I. K. D. D. Archive, “KDD Cup 1999 Data,” 1999. [Online]. Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html
[29] Hassan, “NSL-KDD Dataset,” 2021. [Online]. Available: https://www.kaggle.com/datasets/hassan06/nslkdd
[30] N. Moustafa and J. Slay, “UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems (UNSW-NB15 Network Data Set),” in *Military Communications and Information Systems Conference (MilCIS)*, IEEE, 2015.
[31] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, “Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization,” in *Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP)*, 2018, pp. 108–116.
[32] U. of the Aegean, “Aegean Wireless Intrusion Dataset (AWID),” 2025. [Online]. Available: https://icsdweb.aegean.gr/awid/
[33] J. Gonzalez, R. Gomez, J. M. Blasco, and R. A. Carvajal, “CSIC 2010 HTTP Dataset,” 2010.
[34] N. Koroniotis, N. Moustafa, E. Sitnikova, and J. Slay, “Towards the Development of Realistic Botnet Dataset in the Internet of Things for Network Forensic Analytics: Bot-IoT Dataset,” *Future Generation Computer Systems*, vol. 100, pp. 779–796, 2019.
[35] D. Kwon, H. Kim, J. Kim, S. C. Suh, I. Kim, and K. J. Kim, “A survey of deep learning-based network anomaly detection,” *Cluster Comput*, vol. 22, pp. 949–961, 2019.
[36] O. Faker and E. Dogdu, “Intrusion detection using big data and deep learning techniques,” in *Proceedings of the 2019 ACM Southeast Conference*, 2019, pp. 86–93.
[37] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, “Deep learning approach for intelligent intrusion detection system,” *IEEE Access*, vol. 7, pp. 41525–41550, 2019.
[38] Z. Ahmad, A. S. Khan, C. W. Shiang, J. Abdullah, and F. Ahmad, “Network intrusion detection system: A systematic study of machine learning and deep learning approaches,” *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 1, p. e4150, 2021.

[39] S. M. Kasongo and Y. Sun, “A deep learning method with wrapper based feature extraction for wireless intrusion detection system,” *Comput Secur*, vol. 92, p. 101752, 2020.
[40] Z. Tian, C. Luo, J. Qiu, X. Du, and M. Guizani, “A distributed deep learning system for web attack detection on edge devices,” *IEEE Trans Industr Inform*, vol. 16, no. 3, pp. 1963–1971, 2019.
[41] M. S. Elsayed, N.-A. Le-Khac, S. Dev, and A. D. Jurcut, “Ddosnet: A deep-learning model for detecting network attacks,” in *2020 IEEE 21st International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2020, pp. 391–396.
[42] L. Liu, P. Wang, J. Lin, and L. Liu, “Intrusion detection of imbalanced network traffic based on machine learning and deep learning,” *IEEE Access*, vol. 9, pp. 7550–7563, 2020.
[43] M. M. Hassan, A. Gumaei, A. Alsanad, M. Alrubaian, and G. Fortino, “A hybrid deep learning model for efficient intrusion detection in big data environment,” *Inf Sci (N Y)*, vol. 513, pp. 386–396, 2020.
[44] M. Roopak, G. Y. Tian, and J. Chambers, “Deep learning models for cyber security in IoT networks,” in *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)*, 2019, pp. 452–457.
[45] M. M. Shtayat, M. K. Hasan, R. Sulaiman, S. Islam, and A. U. R. Khan, “An explainable ensemble deep learning approach for intrusion detection in industrial internet of things,” *IEEE Access*, vol. 11, pp. 115047–115061, 2023.
[46] S. Yuan and X. Wu, “Deep learning for insider threat detection: Review, challenges and opportunities,” *Comput Secur*, vol. 104, p. 102221, 2021.
[47] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, and S. Venkatraman, “Robust intelligent malware detection using deep learning,” *IEEE Access*, vol. 7, pp. 46717–46738, 2019.
[48] A. Al-Abassi, H. Karimipour, A. Dehghantanha, and R. M. Parizi, “An ensemble deep learning-based cyber-attack detection in industrial control system,” *IEEE Access*, vol. 8, pp. 83965–83973, 2020.
[49] M. A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, “Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study,” *Journal of Information Security and Applications*, vol. 50, p. 102419, 2020.
[50] S. A. Salloum, M. Alshurideh, A. Elnagar, and K. Shaalan, “Machine learning and deep learning techniques for cybersecurity: a review,” in *The International Conference on Artificial Intelligence and Computer Vision*, 2020, pp. 50–57.
[51] D. S. Berman, A. L. Buczak, J. S. Chavis, and C. L. Corbett, “A survey of deep learning methods for cyber security,” *Information*, vol. 10, no. 4, p. 122, 2019.
[52] G. D. L. T. Parra, P. Rad, K.-K. R. Choo, and N. Beebe, “Detecting Internet of Things attacks using distributed deep learning,” *Journal of Network and Computer Applications*, vol. 163, p. 102662, 2020.
[53] G. Thamilarasu and S. Chawla, “Towards deep-learning-driven intrusion detection for the internet of things,” *Sensors*, vol. 19, no. 9, p. 1977, 2019.
[54] A. R. Luca *et al.*, “Impact of quality, type and volume of data used by deep learning models in the analysis of medical images,” *Inform Med Unlocked*, vol. 29, p. 100911, 2022, doi: 10.1016/j.imu.2022.100911.
[55] M. B. Patel, “Real-Time Violence Detection Using CNN-LSTM.”
[56] F. Croce and M. Hein, “Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks,” Mar. 2020, [Online]. Available: http://arxiv.org/abs/2003.01690
[57] P. Calle *et al.*, “Integration of nested cross-validation, automated hyperparameter optimization, high-performance computing to reduce and quantify the variance of test performance estimation of deep learning models.”